

BOX

B0XX v1.00 *Super Smash Bros. Melee* Instruction Manual

Last Updated: 03/12/2019

Table of Contents

1. Introduction
2. Basic Functionality
 - 2.1. Button Layout
 - 2.2. Recommended Technique
3. The Coordinate Plane
 - 3.1. Tilt and Smash
 - 3.2. 50° Line
 - 3.3. Special Moves
 - 3.4. Out-of-Shield Options
4. Analog Stick Behavior
 - 4.1. Un-Modified Inputs
 - 4.2. Override
5. Modifier Buttons
 - 5.1. Modifier X
 - 5.2. Modifier Y
6. Additional Functionality
 - 6.1. Slight DI
 - 6.2. Firefox Angles
 - 6.3. Extended Up-B Angles
7. Shield
 - 7.1. Automatic Shield Tilt
 - 7.2. Manual Shield Tilt
8. Airdodge
 - 8.1. Wavedash
 - 8.2. Home Row Upwards Airdodge
9. Other Functionality
 - 9.1. Modifier X/Y + Both Horizontal Cardinals
 - 9.2. Angled F-Smash
 - 9.3. D-Pad
 - 9.4. Nunchuk
10. Nerfs

10.1. Accidental Side-B

10.2. Smash DI

10.3. Pivot Tilts

1. Introduction

Developed by Aziz “Hax\$” Al-Yami between December 2016 and March 2019, the B0XX is an ergonomic alternative to the Nintendo Gamecube controller primarily intended for *Super Smash Bros. Melee*, but also compatible with other titles. With its all-digital interface, the B0XX allows *Melee* enthusiasts to enjoy their favorite fast-paced video game while preserving their hand health.

On a software level, the B0XX is thoroughly optimized with two goals in mind: game balance and efficiency. The former goal refers to the fact that the B0XX has been designed *around* the Gamecube controller, rather than with the intention of superseding the Gamecube controller as a viable option for competitors. Certain in-game techniques within *Melee* have deliberately been weakened or excluded from the B0XX in order to match the “strength” of the Gamecube controller as closely as possible with an otherwise superior input method. Efficiency, the latter goal, refers to the fact that a high quality of life has been prioritized across the board: the B0XX’s button arrangement is ideal for the arsenal of techniques available within *Melee*, while the act of aiming the B0XX’s analog directions has been made as intuitive as can be.

Currently, the B0XX is legal at *Melee*’s most prestigious international events, including *Genesis*, *The Big House*, *Super Smash Con*, *Get On My Level*, *Shine*, and more. We hope that as the competitive *Melee* scene continues to evolve, the B0XX goes on to be a testament to the game’s legacy and the love we share for it.

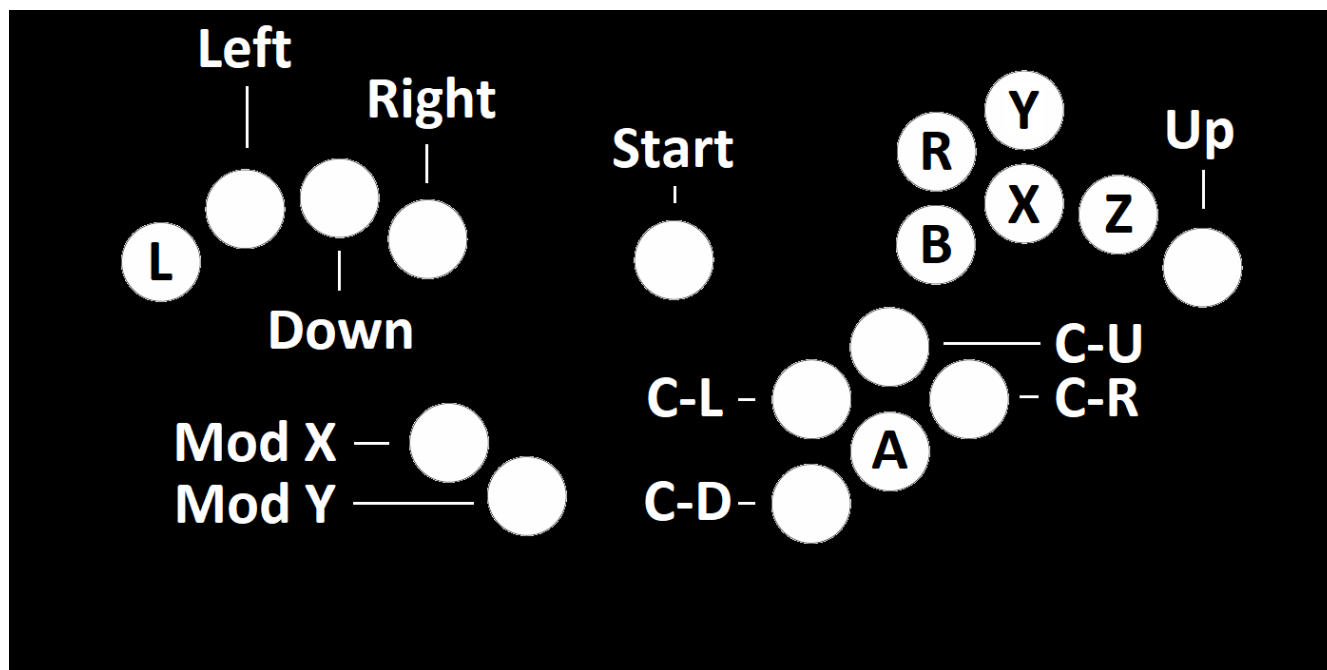
Aziz “Hax\$” Al-Yami (Lead Design)

Stephen “Streamlord” Kasmir (Production and Engineering)

Kyle “Simple” McDowell (Engineering and Software)

2. Basic Functionality

2.1. Button Layout



The B0XX's Left, Down, and Right buttons are assigned to the middle three fingers on your left hand to allow for precise control of your character. L, the B0XX's primary shield trigger, is allocated to your left hand as well. This allows you to comfortably perform out-of-shield actions such as aerial attacks, shield-grabs, and wavedashes with your right hand. Finally, Up, *Melee*'s least important analog direction, is assigned to your right pinky finger as to be accessible at all times (without having to shift your left hand).

Continuing with the right half of the controller, the B and R buttons are each paired with a jump button (X and Y respectively) so that airborne special moves (X → B) and wavedashes (Y → R) can be performed comfortably. Z, which is useful for L-cancelling and grabbing, rounds out the home row of buttons. The A button and the 4 C-stick directions are then situated next to each other so that all five aerial attacks feel similar to perform (X → A or C).

Lastly, Modifier X and Modifier Y, which allow for advanced directional control, will be explained in Chapter 5.

2.2. Recommended Technique

For the sake of comfort and/or efficiency, it is recommended (or in some cases, required) that the following in-game techniques are performed using specific button inputs on the B0XX:

Aerial Attacks: X → A or C

Airborne Special Moves: X → B

Jump-Cancel Grab: X → Z

Tech: L

Wall-Jump Tech: X → L

L-Cancel: Z

Shield (able to Roll, Spottedodge, Tap Jump, and Shield Drop)*: L

Tilted Shield (unable to Roll, Spottedodge, Tap Jump, and Shield Drop)*: R

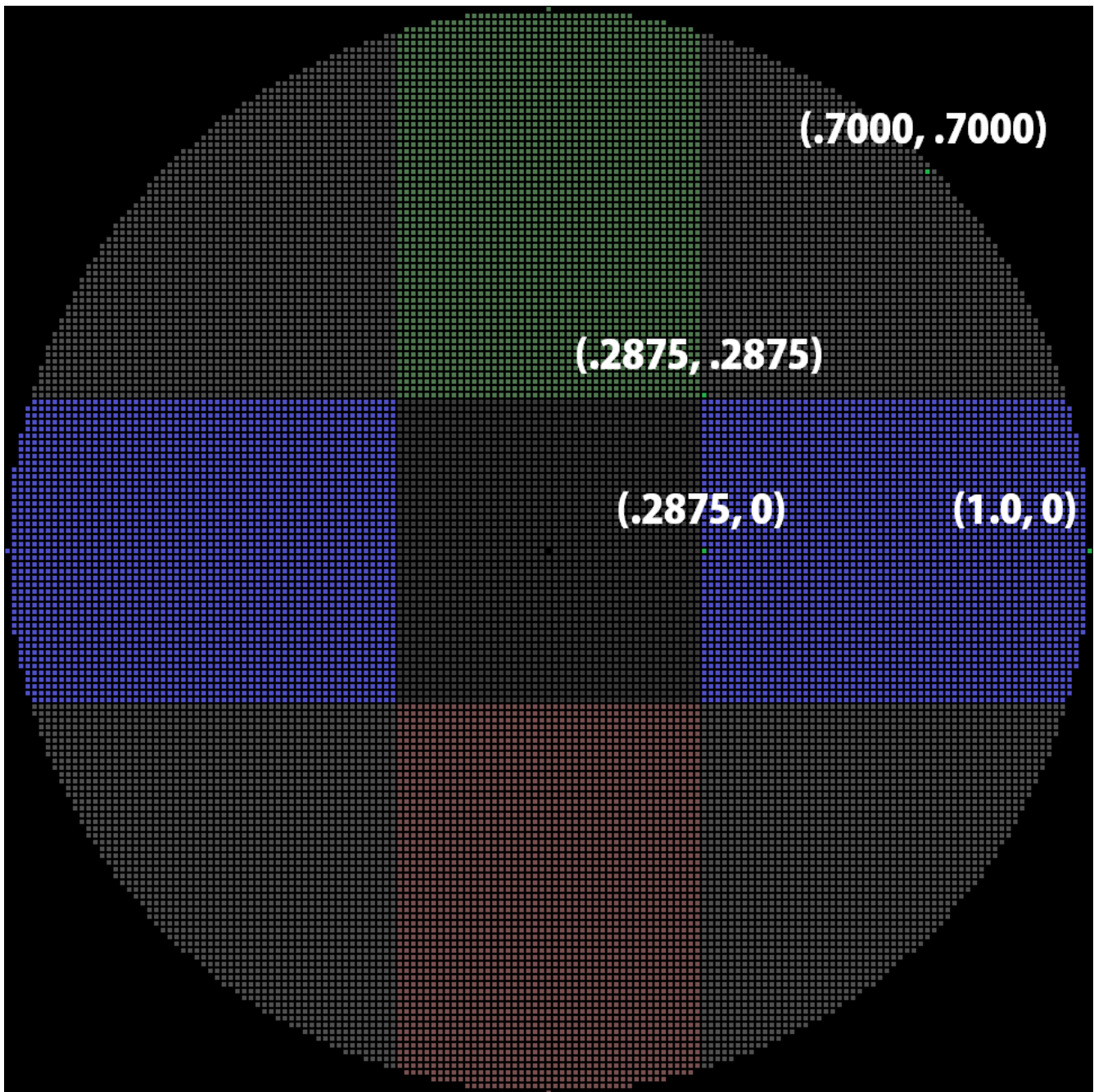
Wavedash (airdodge downwards)**: Y → R

Airdodge Upwards**: L

*The L and R buttons are programmed to generate a shield that is capable of performing out-of-shield actions and a shield that is *incapable* of performing out-of-shield actions respectively. Depending on the in-game situation, either one of these shields can be appropriate. See Chapter 7 for more information.

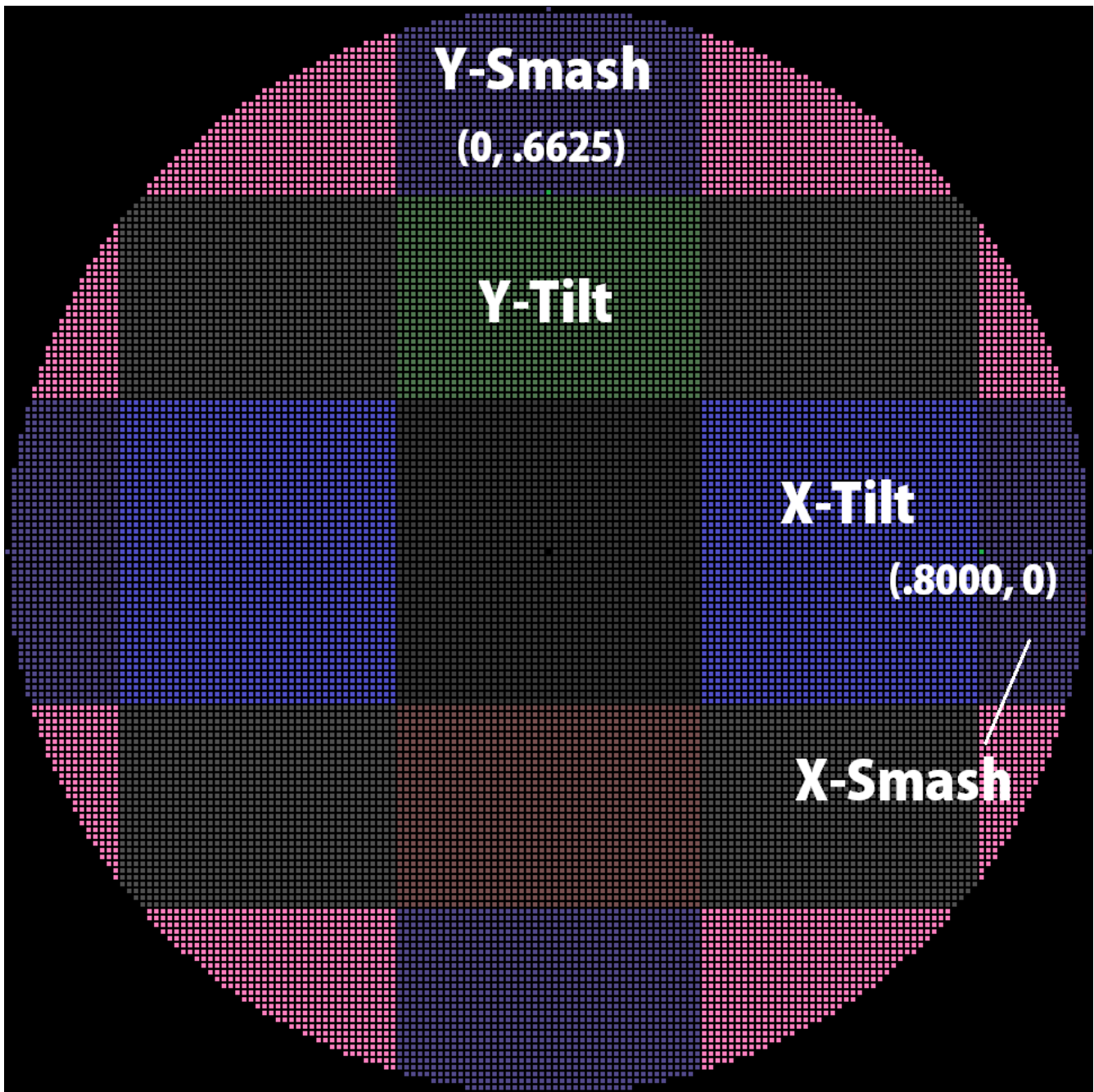
**The discrepancy between airdodging upwards and downwards has to do with the Up button being in an ideal location for being pressed alongside the L button, but not the R button. See Chapter 8 for more information.

3. The Coordinate Plane



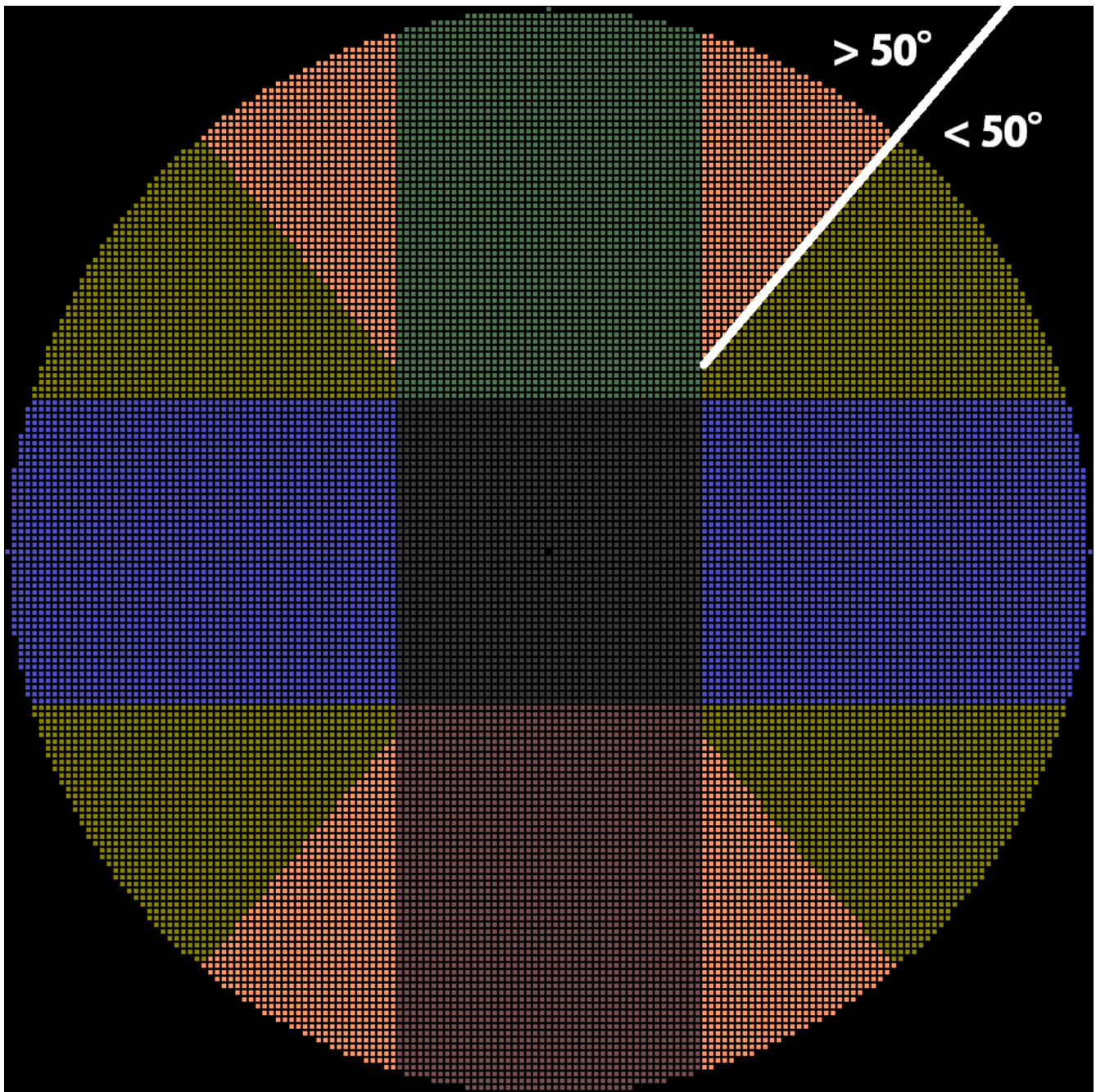
Since the B0XX does not have access to every coordinate on a Gamecube controller's analog stick, a basic overview of the coordinate plane will be helpful for understanding why the coordinates on this controller were chosen. The above diagram showcases the 9 sections of the coordinate plane (dead zone, 4 cardinals, 4 quadrants), as well as the minimum/maximum cardinal vectors and the minimum/maximum 45 degree vectors.

3.1. Tilt and Smash

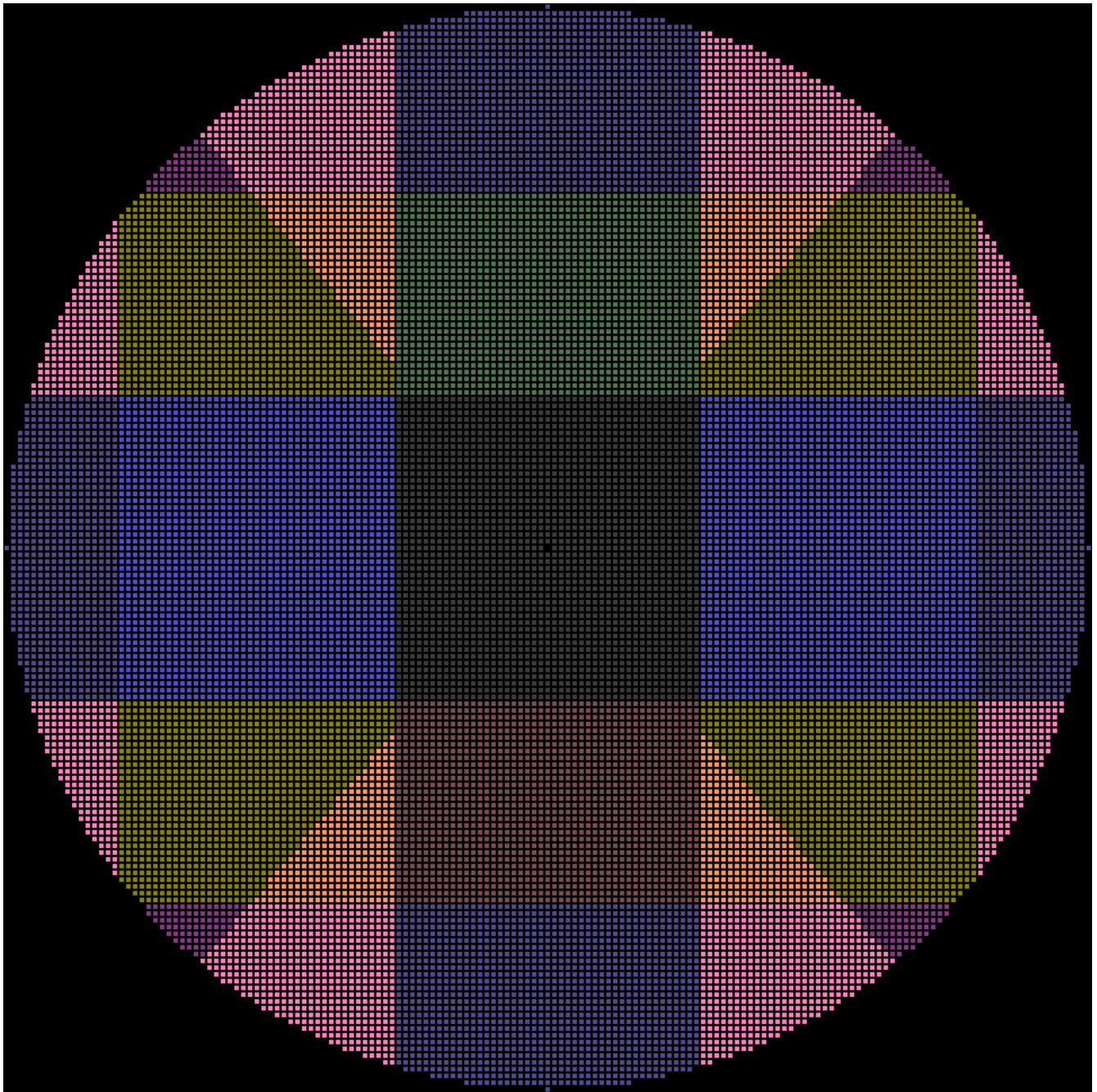


Along the X and Y-axes, there are [different] cut-offs that separate “Tilt” territory from “Smash” territory. The differences between these territories are not limited to Tilt and Smash attacks, but also actions such as Walk/Dash along the X-axis and Tap Jump along the Y-axis. X .8000 and Y .6625, the first values that activate Smash on either axis, are pictured above.

3.2. 50° Line

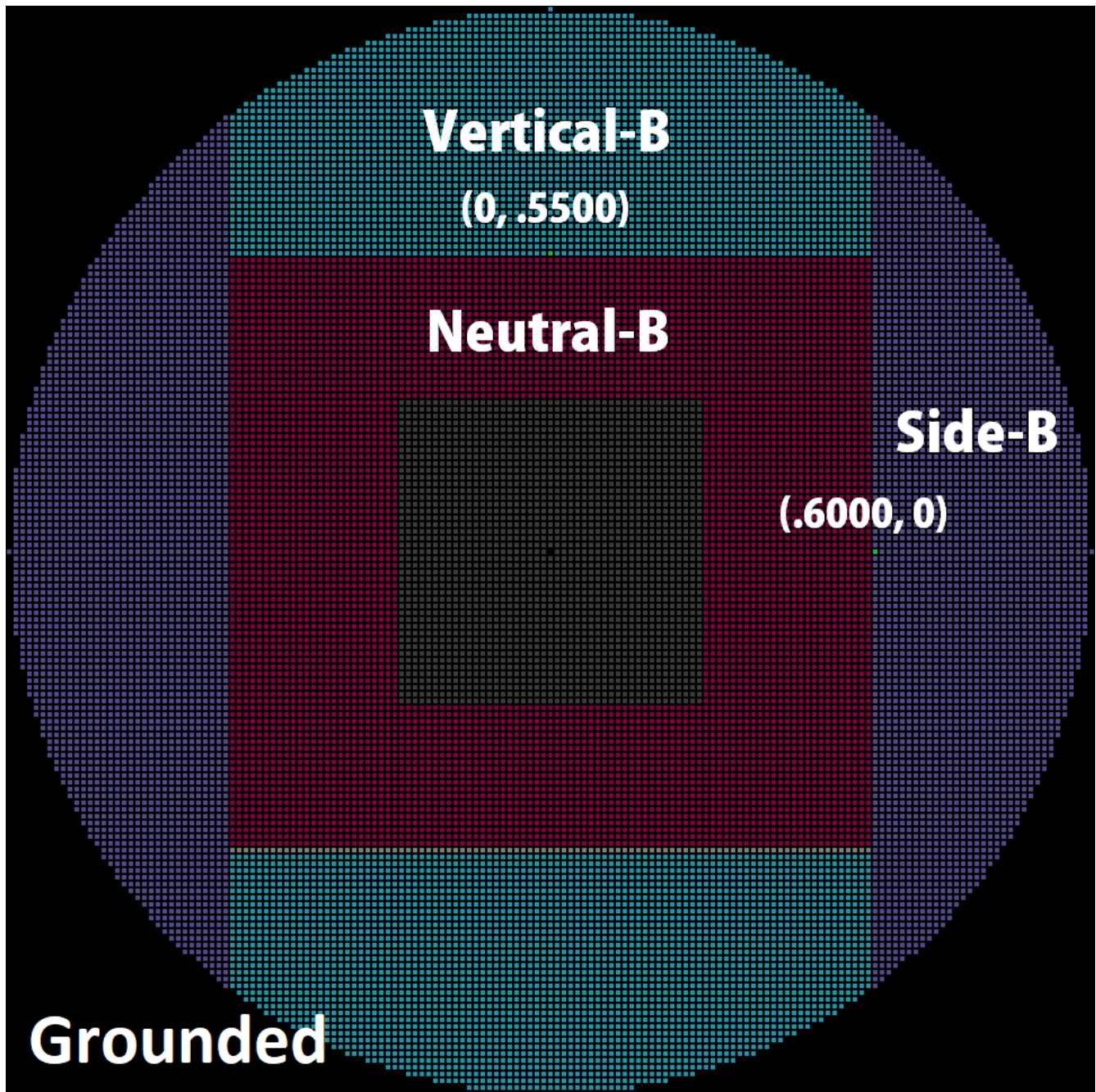


Within the quadrants, there is a 50° Line that distinguishes various actions. Most notably, $< 50^\circ$ territory is useful for angled F-tilt and getting up from the ledge, while $> 50^\circ$ territory is useful for turnaround U-tilt/D-tilt and falling from the ledge.

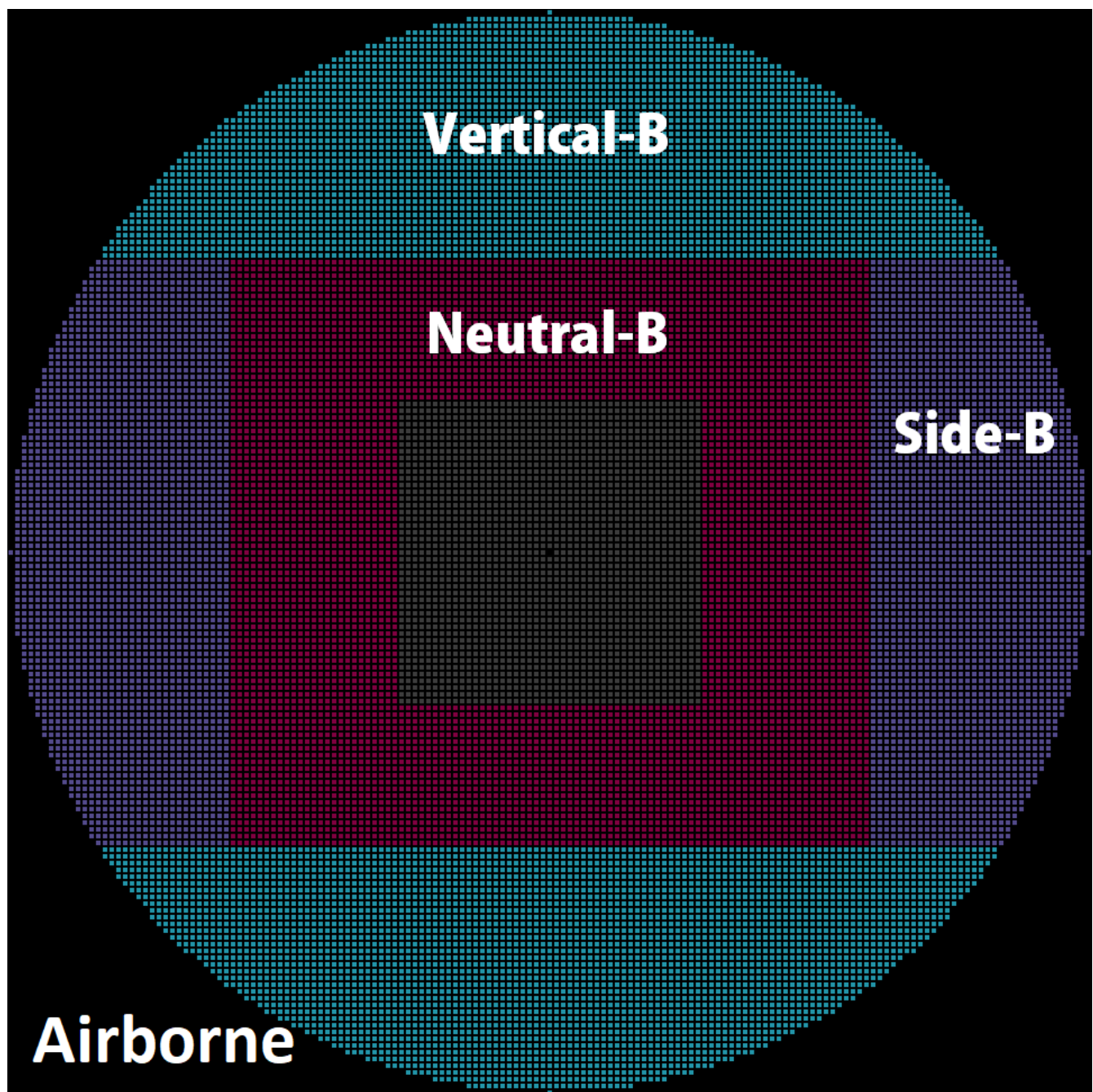


In case you were curious, here is a diagram showing what the Tilt/Smash zones + 50° Line look like when overlapped.

3.3. Special Moves

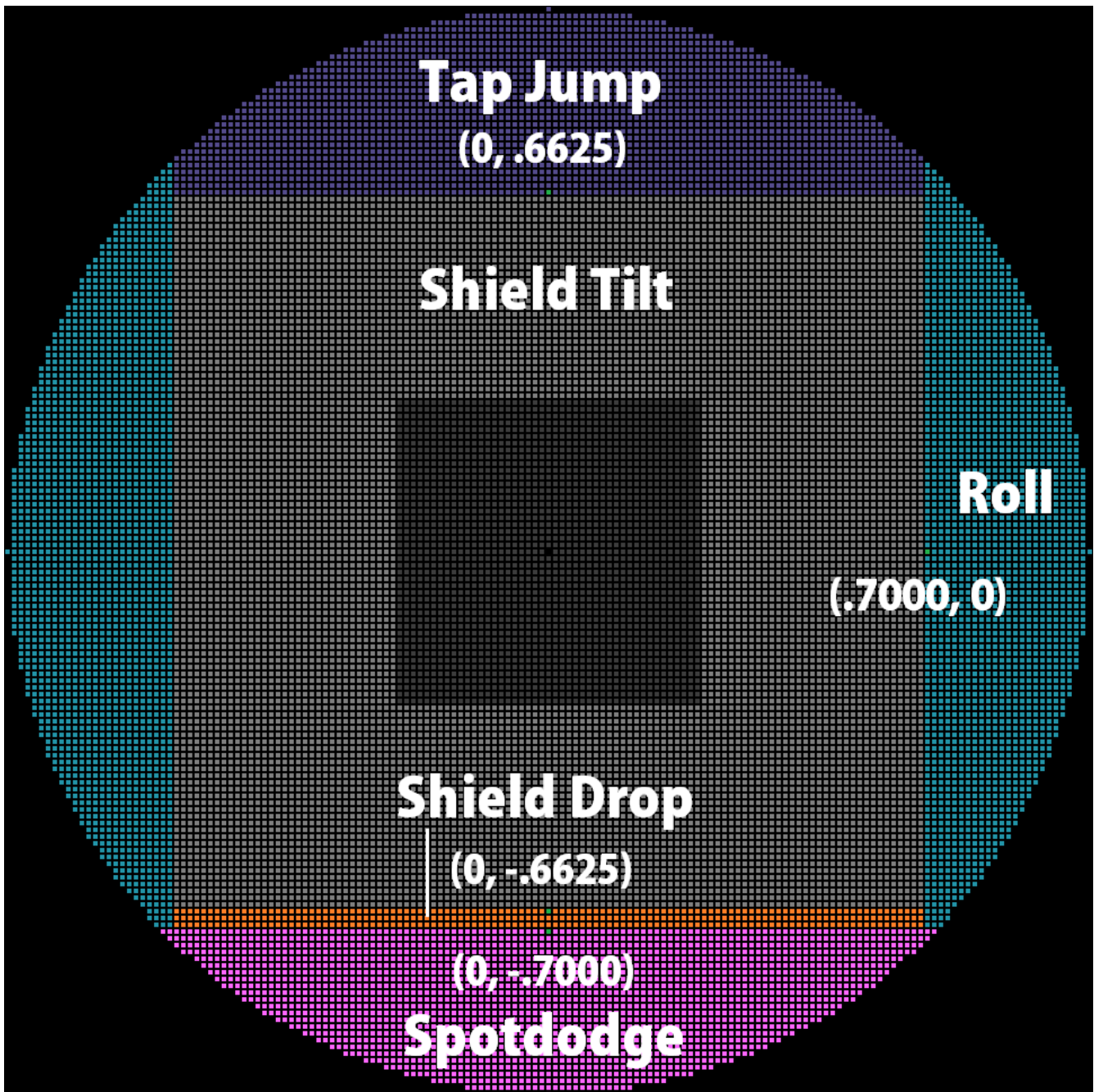


When your character is grounded, a neutral-B can be performed by pointing in either red territory or the center dead zone, a side-B can be performed by pointing in purple territory, and a vertical-B can be performed by pointing in blue territory.



When your character is airborne, neutral-B territory remains the same, but the corners of the coordinate plane are swapped; what was formerly side-B territory becomes vertical-B territory.

3.4. Out-of-Shield Options

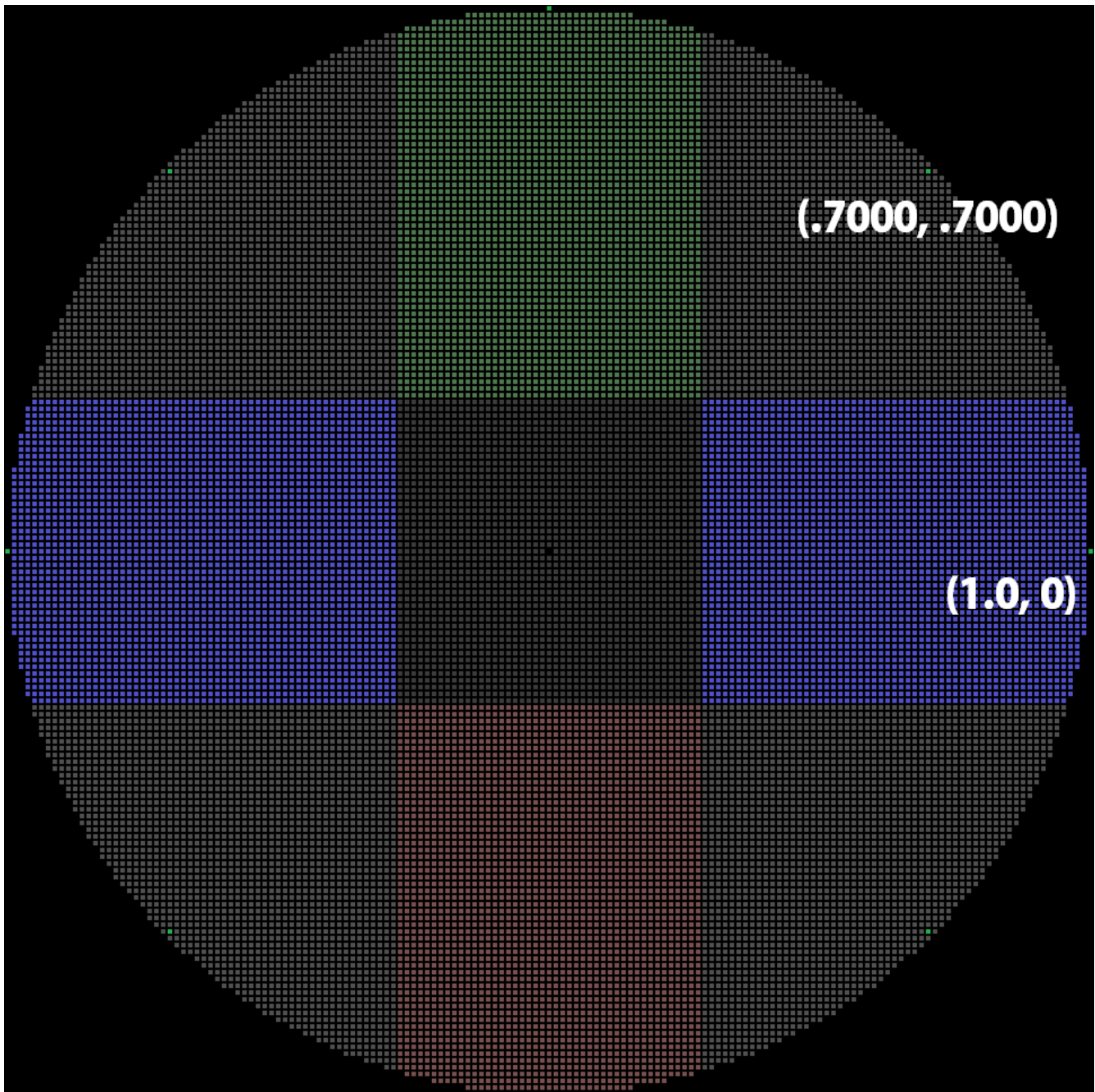


When you are shielding, Roll and Spotdodge can be performed by pointing at $\Rightarrow .7000$ in the appropriate direction. Tap Jump still requires a Smash input along the Y-axis ($\Rightarrow .6625$), while Shield Drop requires a very precise Y-axis input of $-.6625$, $-.6750$, or $-.6875$ (hence the need for Shield Drop notches!).

If none of these conditions are met, shield can be tilted within the gray area.

4. Analog Stick Behavior

4.1. Un-Modified Inputs



On their own (without the use of Modifier X, Modifier Y, or any other button on the B0XX), the 4 analog stick directions will pinpoint 1.0 cardinal vectors and X .7000 Y .7000 (45°) diagonal vectors. This behavior applies to the 4 C-stick directions as well.

4.2. Override

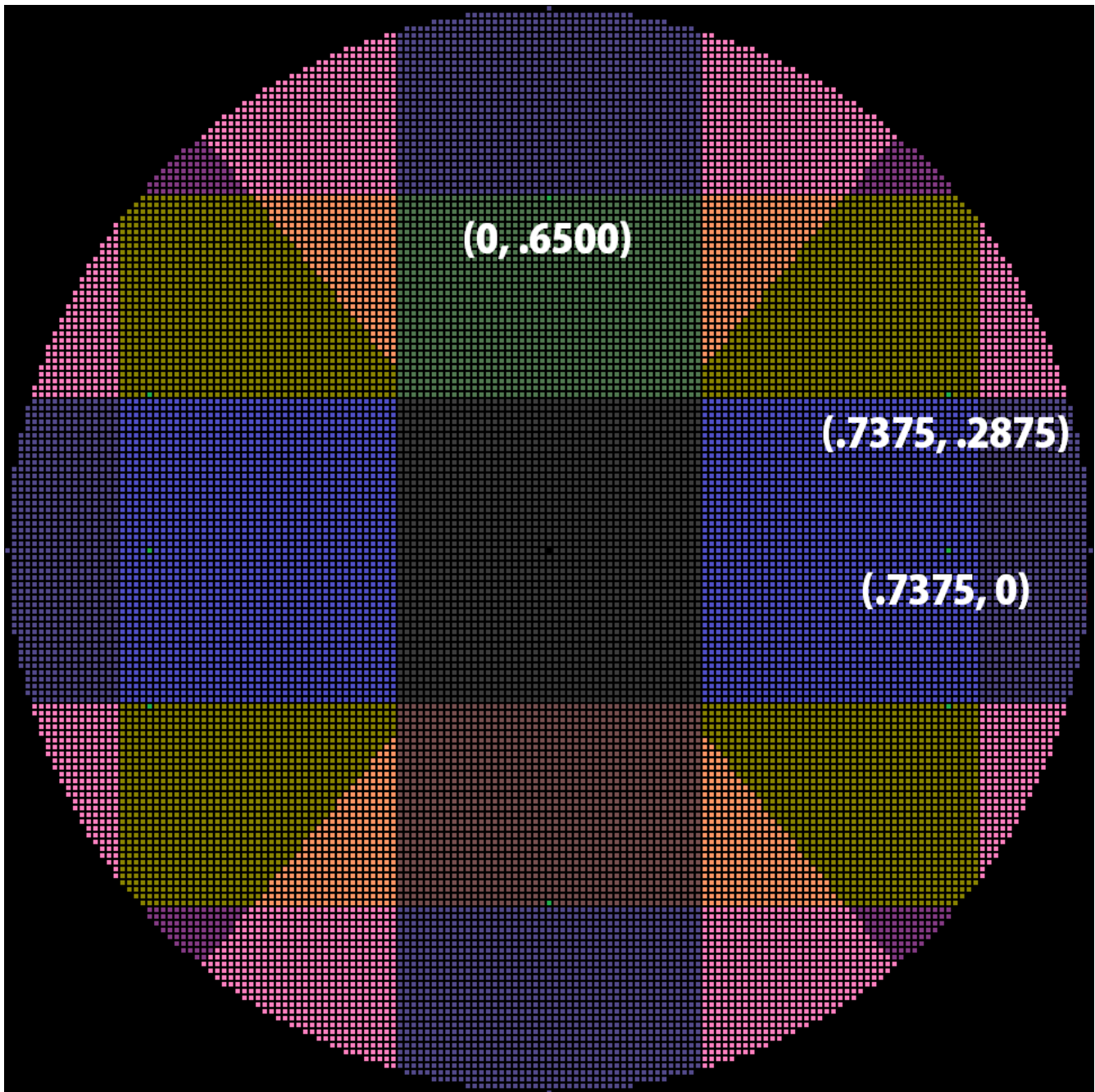
The B0XX handles opposite cardinal inputs (i.e. Left → Right) by overriding the previously held direction with the newly chosen direction. It is crucial that the B0XX's analog stick behaves as such, as this allows the B0XX user to dash-dance (dash left and right quickly) without having to worry about newly chosen directions not being registered.

If, however, opposite cardinal inputs are held simultaneously and the most recently chosen direction is released (i.e. Left → Right → release Right), the original direction will **not** reactivate. This system is intended to most closely resemble the act of releasing a Gamecube controller's analog stick, which would cause it to return to its centerpoint, *not* the opposite cardinal direction.

The behavior described within this section applies to the B0XX's C-stick (and D-pad; see Section 9.3) as well.

5. Modifier Buttons

5.1. Modifier X



Horizontals: X .7375
Verticals: Y .6500
Quadrants: X .7375 Y .2875

Modifier buttons are akin to Shift keys on a keyboard; on their own, modifier buttons won't produce any inputs, but in conjunction with the 4 analog stick directions, they will adjust your inputs from the ones shown in Section 4.1.

Modifier X, the B0XX's primary modifier button, is named after its ability to **hug the X-axis within the quadrants**. But before Modifier X's quadrant modifications are explained, its cardinal modifications will be.

When held in conjunction with a horizontal cardinal input, Modifier X will shift you from X 1.0 to X .7375. This coordinate is notable for being **the greatest X-value in the game that does not break Teeter (the mechanic that prevents your character from falling off the end of the stage)**. X .7375 is most often useful for **Walking, F-tilting, and slight DI'ing** (especially when you are thrown vertically by your opponent).

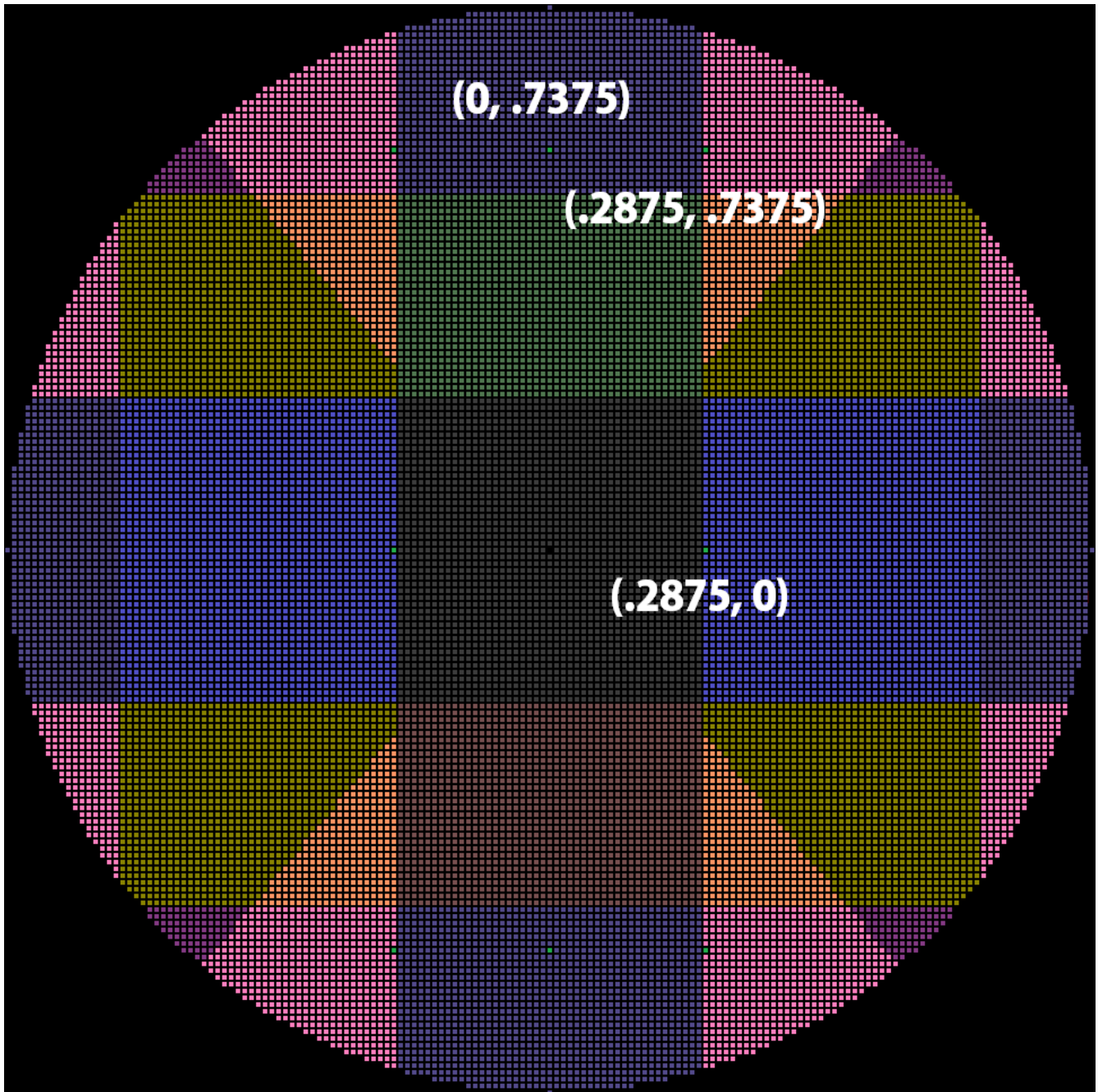
When held in conjunction with a vertical cardinal input, Modifier X will shift you from Y 1.0 to Y .6500, the greatest Y-value within Y-tilt. Typically, Y .6500 is useful for performing a **U-tilt or D-tilt**.

Finally, within the quadrants, Modifier X's coordinates of X .7375 Y .2875 produce a $< 50^\circ$ angle (21.3° to be exact) and allow you to perform an **angled F-tilt**. These coordinates also double as **the shallowest Firefox angle on the B0XX**, which will be elaborated on in Section 6.2.

Not pictured: Modifier X will also modify an L button input to an **analog L 49** input. Doing so can be used to generate a lightshield of the exact strength as a Z button lightshield.

Advanced: When held in conjunction with *both* horizontal cardinal directions (Left + Right simultaneously), Modifier X will *not* cause your X-value to shift from 1.0 to .7375 (it will remain 1.0). Unless you're an experienced B0XX user, don't worry about this feature. If you're an experienced B0XX user... see Section 9.1 for more information.

5.2. Modifier Y



Horizontals: X .2875
Verticals: Y .7375
Quadrants: X .2875 Y .7375

Nearly a mirror image of Modifier X, Modifier Y is aptly named after its ability to **hug the Y-axis within the quadrants**. By pinpointing X .2875

Y .7375, Modifier Y allows you to perform **turnaround U-tilt, but only in buffered situations** such as L-cancel lag or wavedash lag (if you press A in conjunction with X .2875 Y .7375 when your character is actionable, you will perform a U-smash). Additionally, these coordinates form a 68.7° angle, which can be used to **Firefox as steeply as the B0XX permits** (see Section 6.2).

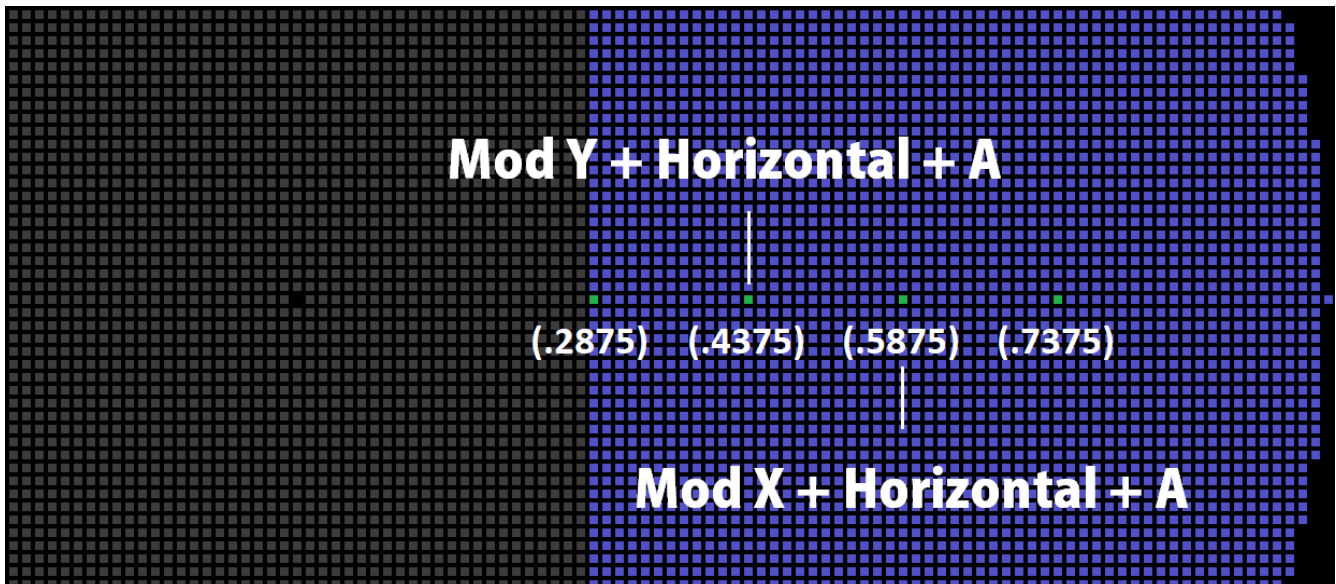
When held in conjunction with a horizontal cardinal input, Modifier Y will shift you from X 1.0 to X .2875. X .2875 is primarily intended to serve as **another slight DI option** for when you are thrown vertically by your opponent.

Not pictured: Just like Modifier X, Modifier Y will modify an L button input to analog L 49.

Advanced: Once again, holding *both* horizontal cardinal directions in conjunction with Modifier Y won't result in your X-value of 1.0 being modified. See Section 9.1.

6. Additional Functionality

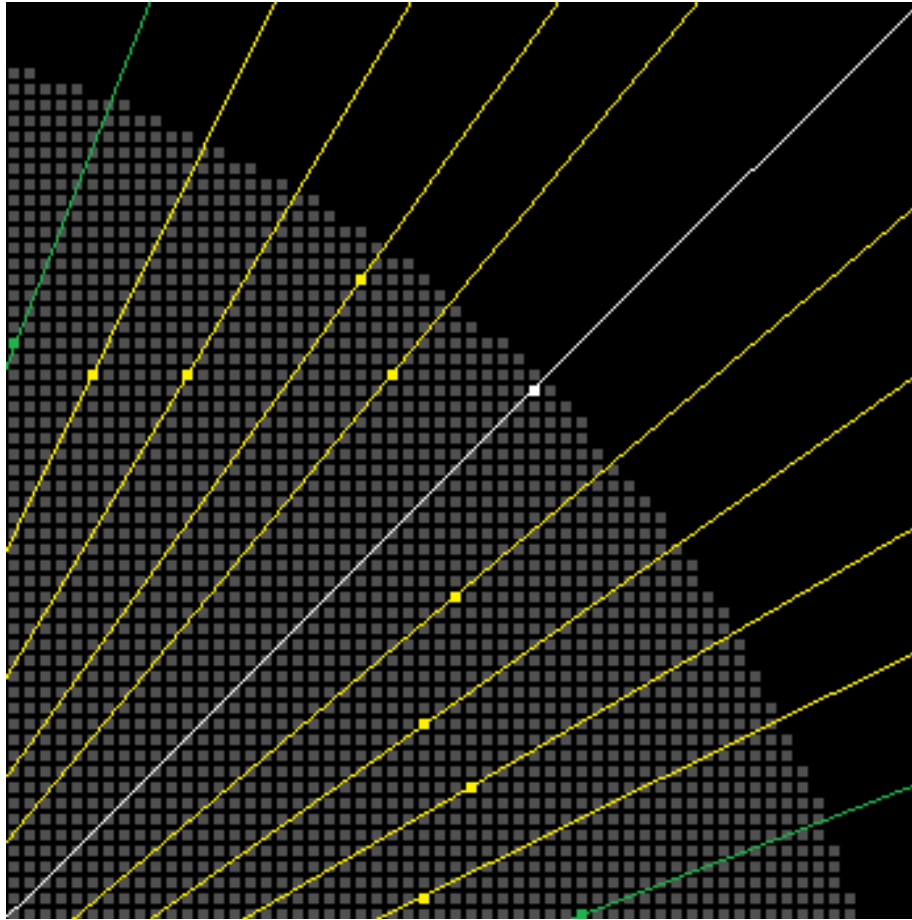
6.1. Slight DI



On their own, the slight DI options shown thus far (X .2875 and .7375) are insufficient; at any level of play, it is necessary for the large gap in between those coordinates to be filled. **X .4375 and .5875 – two additional slight DI options** – are accessible by holding a horizontal direction and the **A button** in conjunction with Modifier X or Y respectively.

6.2. Firefox Angles

This section is only relevant if you are playing Fox/Falco.



White: 45°

Green: 21.3° and 68.7°

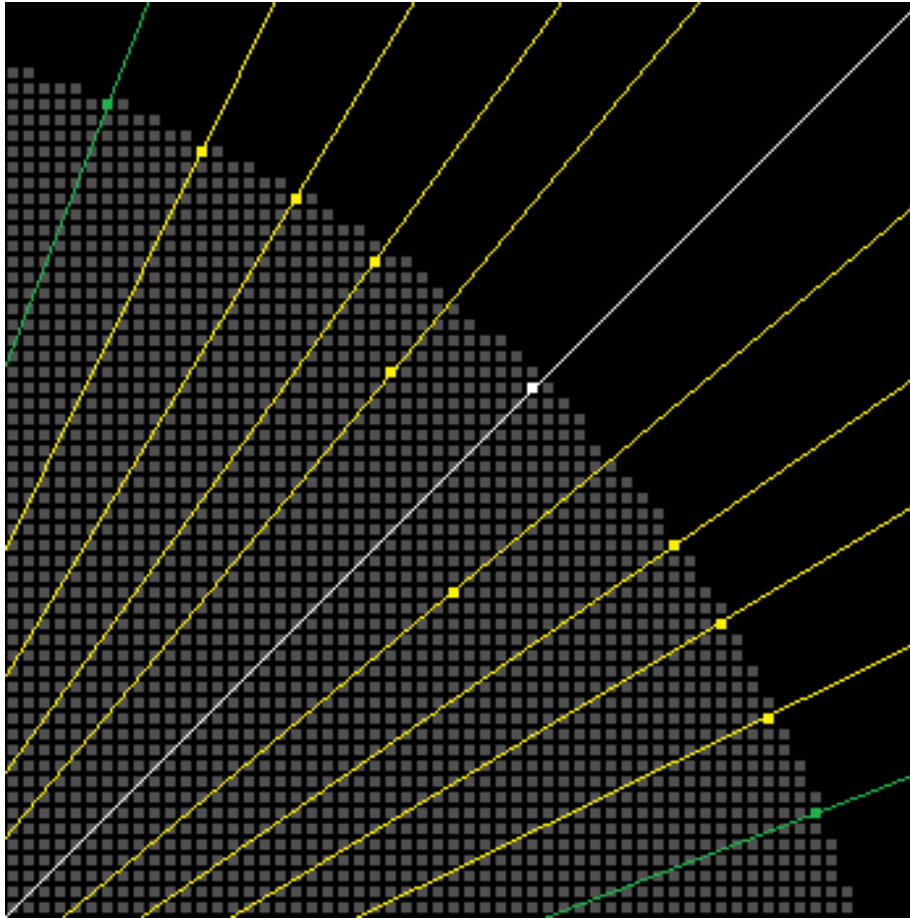
Yellow: Evenly distributed angles in between

Similarly, Modifier X and Y's 21.3° and 68.7° angles (and the un-modified 45° angle) aren't nearly sufficient for recovering as Fox/Falco. By holding each of the **4 C-stick directions** in conjunction with either of the modifier buttons, **4 additional Firefox angles are unlocked**.

Note: These aren't just any Firefox angles; they're the most evenly distributed angles possible in the entire game.

6.3. Extended Up-B Angles

This section is only relevant if you are playing Mewtwo/Pichu/Pikachu/Sheik/Zelda.



Green: 21.4° and 68.6°

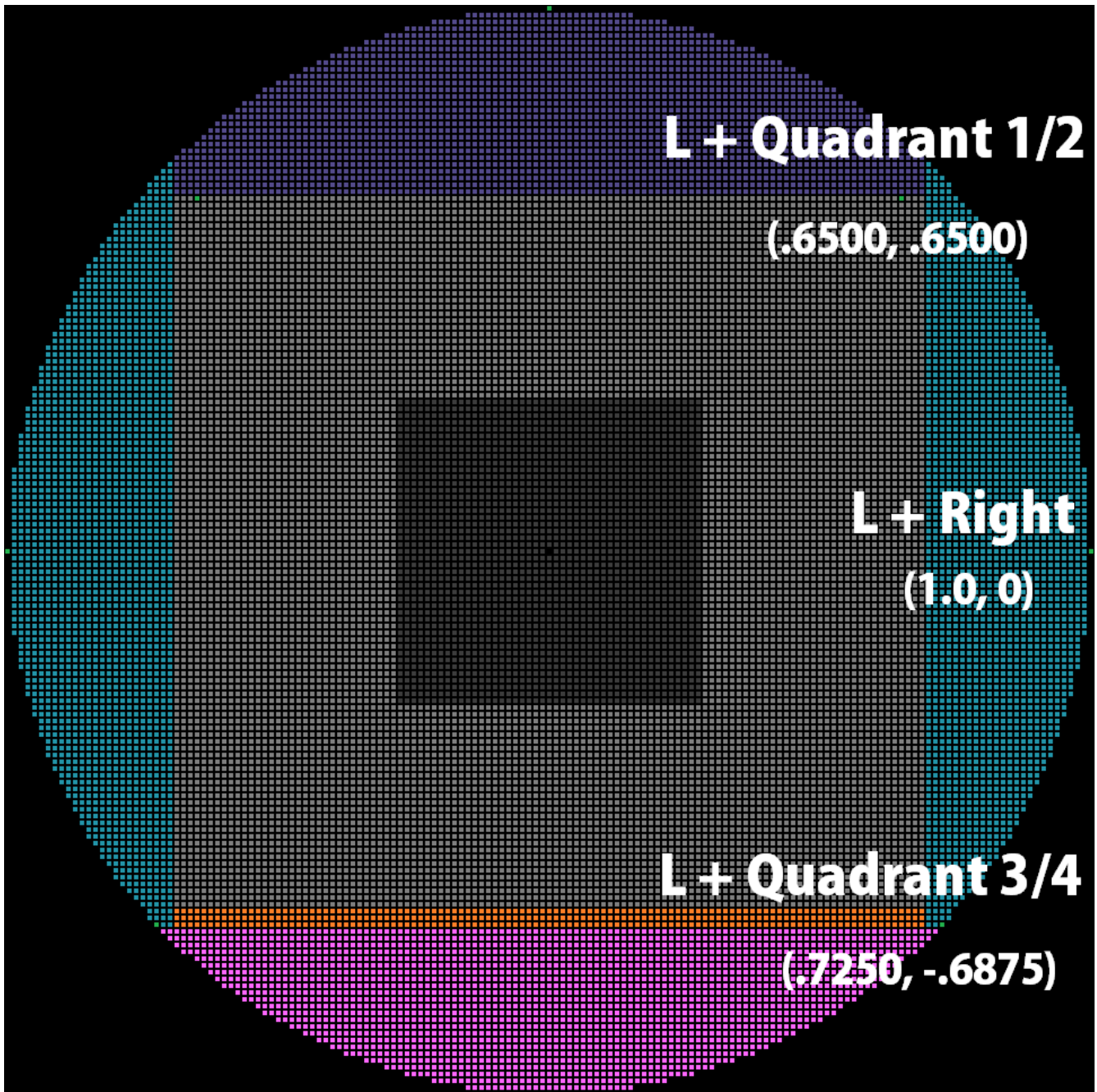
White: 45°

Yellow: Evenly distributed angles in between

Unfortunately, the other 5 characters who can angle their up-B won't travel as far if you use the Firefox coordinates from Section 6.2 because their up-B's are reliant on the magnitude of the vector used to angle them. For this reason, an additional step is required to push Section 6.2's coordinates outwards and towards the rim. To access these higher magnitude coordinates, hold **Modifier X/Y and L** in addition to one of the **4 C-stick directions**. For the non-C angles (pictured in green), hold **Modifier X/Y, L, and B**.

7. Shield

7.1. Automatic Shield Tilt



Horizontals: X 1.0

Verticals: Y 1.0

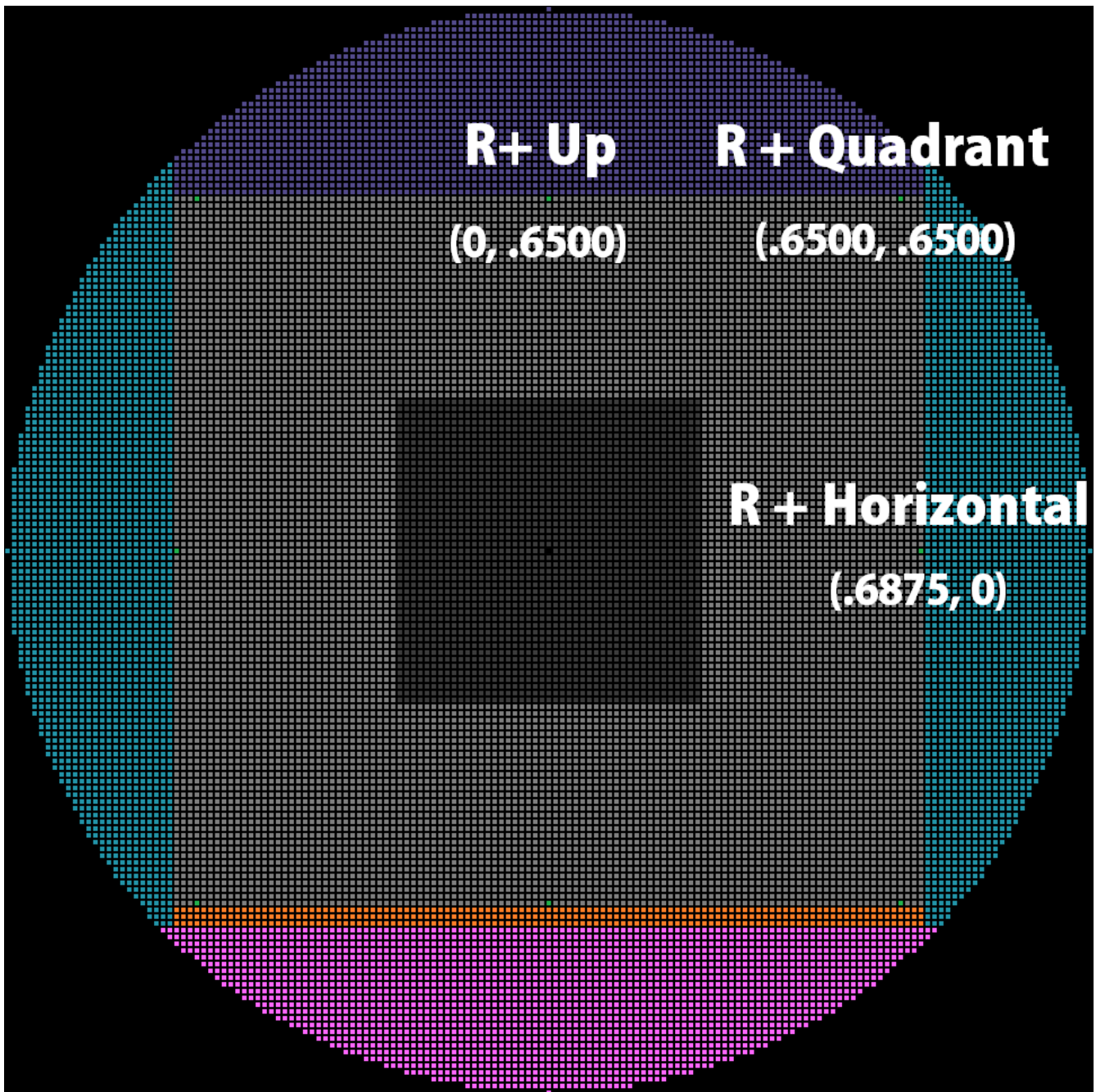
Quadrants 1 and 2: X .6500 Y .6500

Quadrants 3 and 4: X .7250 Y -.6875

The B0XX's primary shield button, L is intended to serve as an all-purpose shield that presents you with the most desirable default options. To accomplish this goal, L causes no alterations to the cardinal analog stick inputs of X 1.0 or Y 1.0; these remain usable for out-of-shield actions such as Roll, Spotdodge, and Tap Jump. In the quadrants, however, L modifies the coordinates X .7000 Y .7000 in favor of coordinates that *don't* Spotdodge or Tap Jump (as these would be redundant), but allow you to **tilt your shield diagonally upwards/downwards** instead. Additionally, in quadrants 3 and 4, L allows you to perform a **shield drop** (regardless of whether you are playing on vanilla *Melee* or UCF). L's modifications are collectively referred to as *Automatic Shield Tilt*.

For clarification, in quadrants 1 and 2, the coordinates X .6500 Y .6500 are chosen in part so that **an upwards airdodge that is performed with L takes place at a 45° angle** (as opposed to an arbitrary #).

7.2. Manual Shield Tilt



Horizontals: X .6875

Verticals: Y .6500

Quadrants 1, 2, 3, and 4: X .6500 Y .6500

R covers L's bases by allowing you to **tilt your shield in any direction without Rolling, Spotdodging, Tap Jumping, or Shield Dropping**. This

feature, known as *Manual Shield Tilt*, is necessary in situations where out-of-shield options are undesirable (such as angling shield downwards to protect your character's feet while standing on a platform).

Advanced: If L and R are held simultaneously, R's shield modifications will take priority. It is recommended that you take advantage of this feature by tilting your shield (with L + R) in a cardinal direction for a few frames, then releasing R, at which point L will allow you to tilt your shield as far outwards as possible (X or Y 1.0) without Rolling, Spotdodging, Tap Jumping, or Shield Dropping. The number of frames you must tilt your shield for varies based on the out-of-shield action you point towards:

Shut off Roll: Hold R for 4 frames

Shut off Spotdodge: Hold R for 4 frames

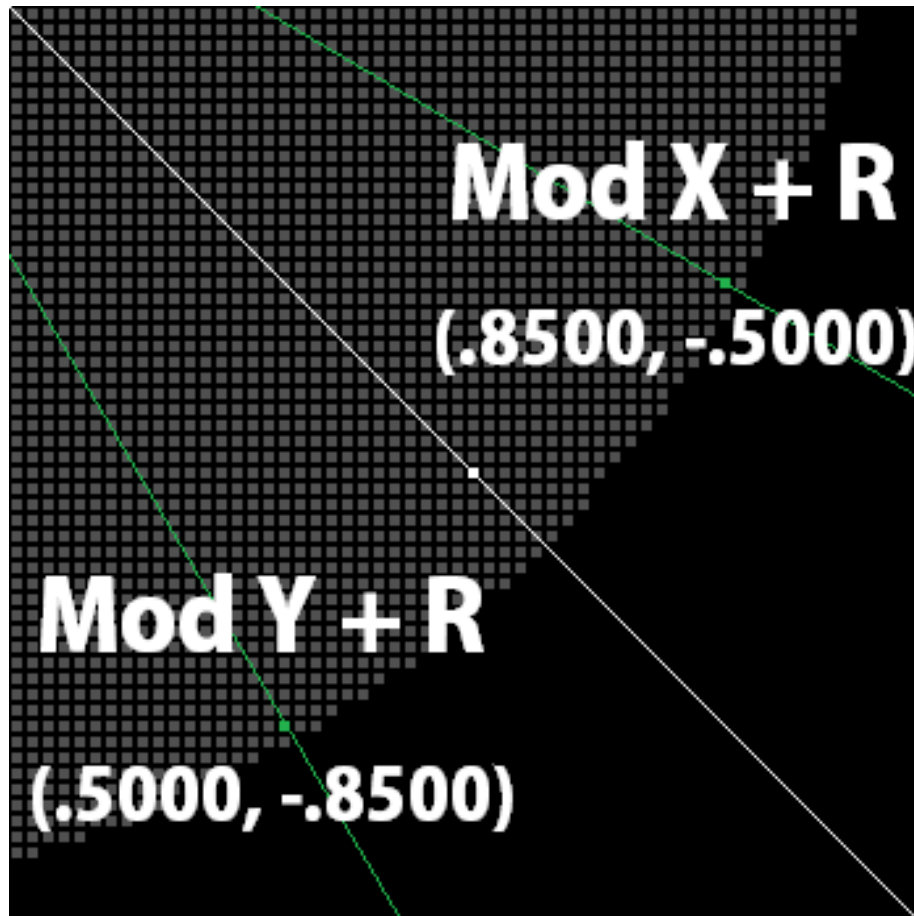
Shut off Tap Jump: Hold R for 4 frames

Shut off Shield Drop: Hold R for 6 frames

Lastly, R's quadrant coordinates of X .6500 Y .6500 are once again chosen so that **an airdodge that is performed with R takes place at a 45° angle.**

8. Airdodge

8.1. Wavedash



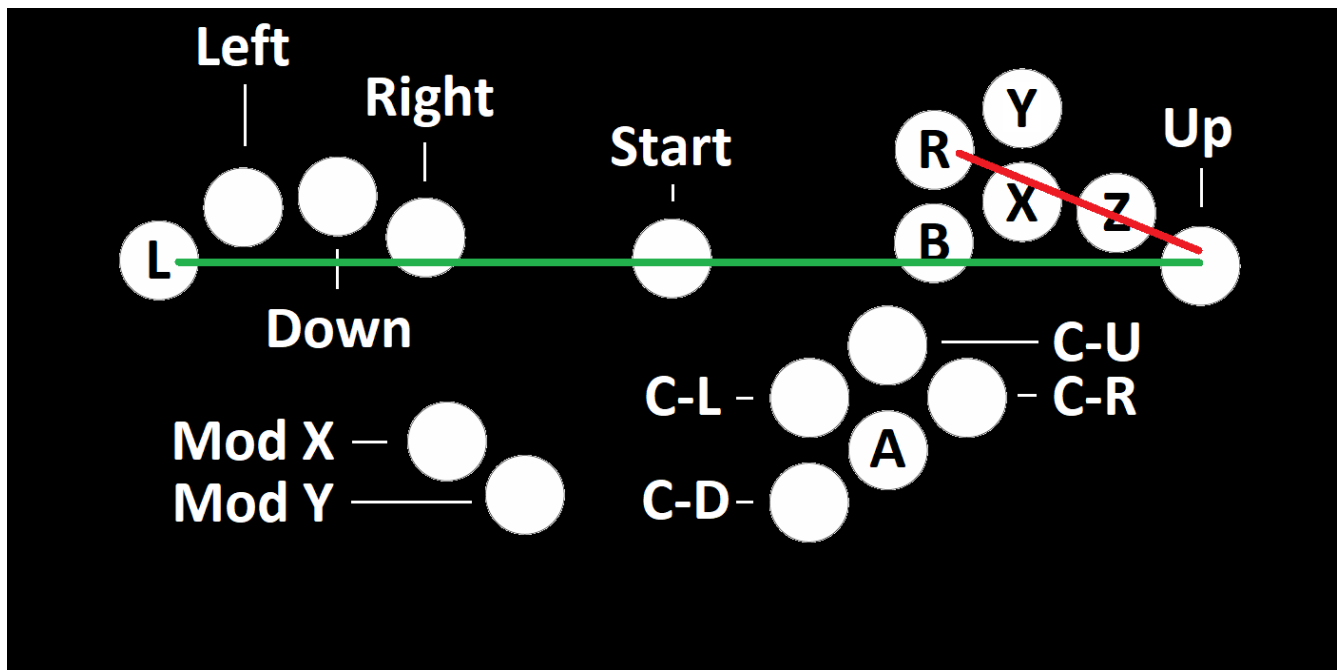
White: X .6500 Y .6500 (45°)

Green: X .8500 Y -.5000 (30.5°) and X .5000 Y -.8500 (59.5°)

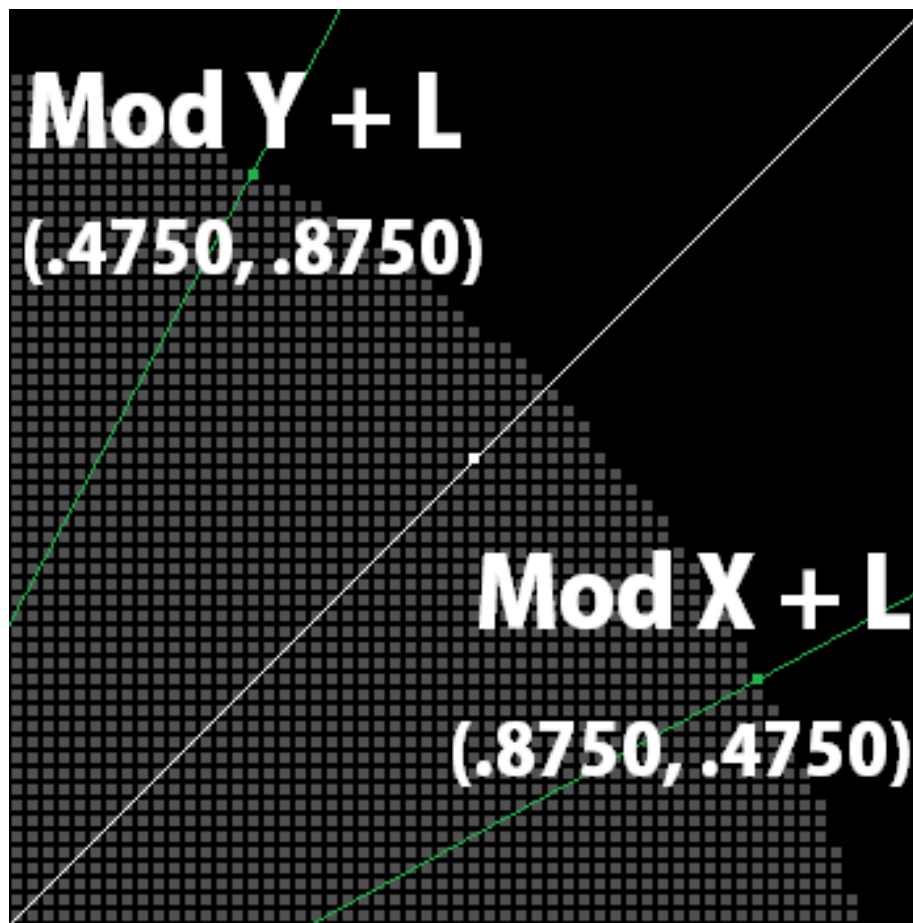
The B0XX's primary airdodge button, R is assigned your right index finger to allow for the swift and aggressive button presses that are often necessary when performing wavedashes. On a software level, **R supports airdodging by *reducing* the steepness of Modifier X/Y's 21.3° and 68.7° angles; R will adjust these to 30.5° and 59.5° respectively.** This deliberate nerf is intended to simulate the increased difficulty a Gamecube controller user faces in airdodging steeply (as compared to up-B'ing steeply) due to the smaller timing window for aiming an airdodge.

As with all of the other coordinates on the B0XX, X .8500 Y -.5000 (30.5°) and X .5000 Y -.8500 (59.5°) are non-arbitrary. After thoroughly examining the impact an airdodge angle has on ledge-dash sequences, these angles proved to be magic #'s for some of *Melee*'s most popular characters. They were finalized on the basis of promoting healthy ledge-dash behavior across the cast, as well as being reasonably reliable angles to pinpoint when airdodging with a Gamecube controller.

8.2. Home Row Upwards Airdodge



The purpose of *Home Row Upwards Airdodge*, the feature covered within this section, is **to allow the B0XX's user to airdodge upwards comfortably**. The R button, unfortunately, cannot provide adequate comfort in performing this technique as it is not located on the same row of buttons as Up (and would force the user to tilt their wrist diagonally if required). Luckily, the L button's location makes for a satisfactory workaround to this issue.



White: X .6500 Y .6500 (45°)

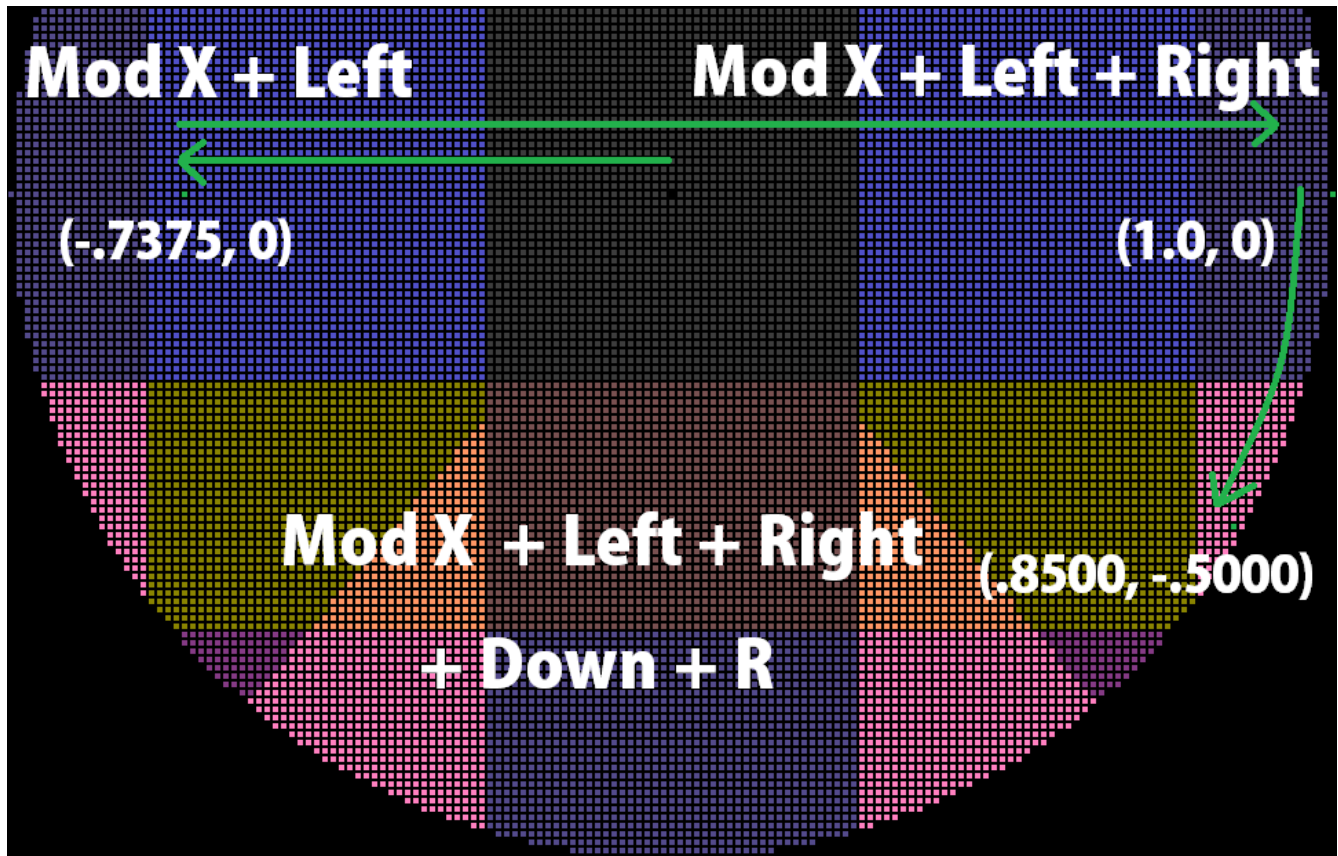
Green: X .8750 Y .4750 (28.5°) and X .4750 Y .8750 (61.5°)

In Chapter 5, it was briefly mentioned that Modifiers X and Y both modified an L input to an analog L 49 (lightshield) input; however, in order to give the B0XX the ability to airdodge upwards with L, **Modifier X and Y's analog L 49 modifications only occur when the analog stick is being pointed horizontally or downwards.** When the analog stick is being pointed upwards, Modifier X and Y cause L to remain a digital (hard press) input, allowing you to airdodge at the angles shown above.

Due to an exploitative technique that can be performed by airdodging at X .5000 Y .8500 (59.5°) with Peach, **the B0XX's upwards airdodges are ever-so-slightly different from its downwards airdodges.** R's upwards airdodges are kept in line with L's not with the intention of actually being used, but for the sake of fully removing this exploit from the controller.

9. Other Functionality

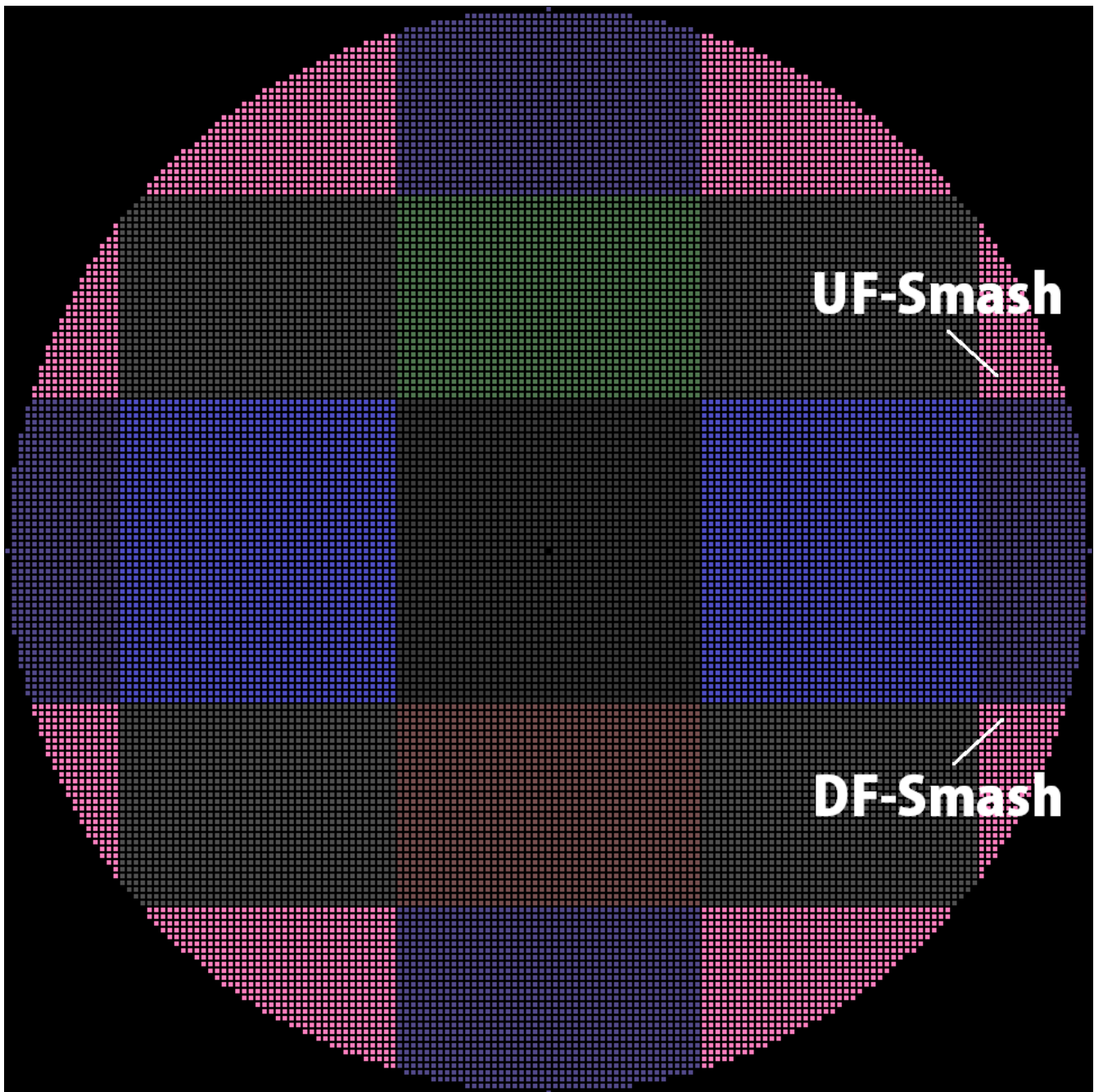
9.1. Modifier X/Y + Both Horizontal Cardinals



In Chapter 5, an inconspicuous, but crucial feature was listed: Modifier X and Y's refusal to modify horizontal analog stick inputs *when Left and Right are held simultaneously*.

This feature was implemented in order to optimize a single in-game scenario: **ledge-dashing**. During a ledge-dash, the **jump trajectory** registered on your character's mid-air jump frame contributes to not only the overall distance your character travels, but in some cases, whether the ledge-dash succeeds or fails altogether! By retaining the best possible jump trajectory value of X 1.0 when both horizontal cardinals are held, Modifiers X and Y allow you to ledge-dash at different airdodge angles *without* arbitrarily stunting your mid-air jump. This benefit can be seen in the above diagram, which demonstrates the player depressing Modifier X, falling from the ledge with Back, jumping with X 1.0 Forward, then airdodging towards the stage at a 30.5° angle.

9.2. Angled F-Smash



With the B0XX's analog stick, it is impossible to pinpoint angled F-smash territory (outside of a few of the Extended Up-B angles from Section 6.3, but those require L to be held). In order to perform these Smash attacks, **hold Modifier X in conjunction with either analog Up or Down (depending on which angled F-smash you want). Then, press C-Left or C-Right to receive an angled F-smash input on the C-stick.**

9.3. D-Pad

To access the B0XX's D-pad, hold down Modifier X and Modifier Y simultaneously. The 4 C-stick directions will then transform into the D-pad.

Note: When Modifier X and Modifier Y are held simultaneously, neither of their analog stick modifications or analog L 49 (lightshield) modifications will work until either Modifier X or Modifier Y is released.

9.4. Nunchuk



The B0XX is compatible with the Nintendo Wii Nunchuk (sold separately). Once plugged in, the Nunchuk will cause the B0XX to hot-swap into *Nunchuk Mode*, during which the 4 analog stick directions, Modifier X and Y, and L are disabled. The Nunchuk's Z button (large button) then behaves as L, while its C button (small button) behaves as analog L 49. *Nunchuk Mode* allows Gamecube controller users to enjoy the B0XX's improved button layout while retaining the analog stick control they are familiar with.

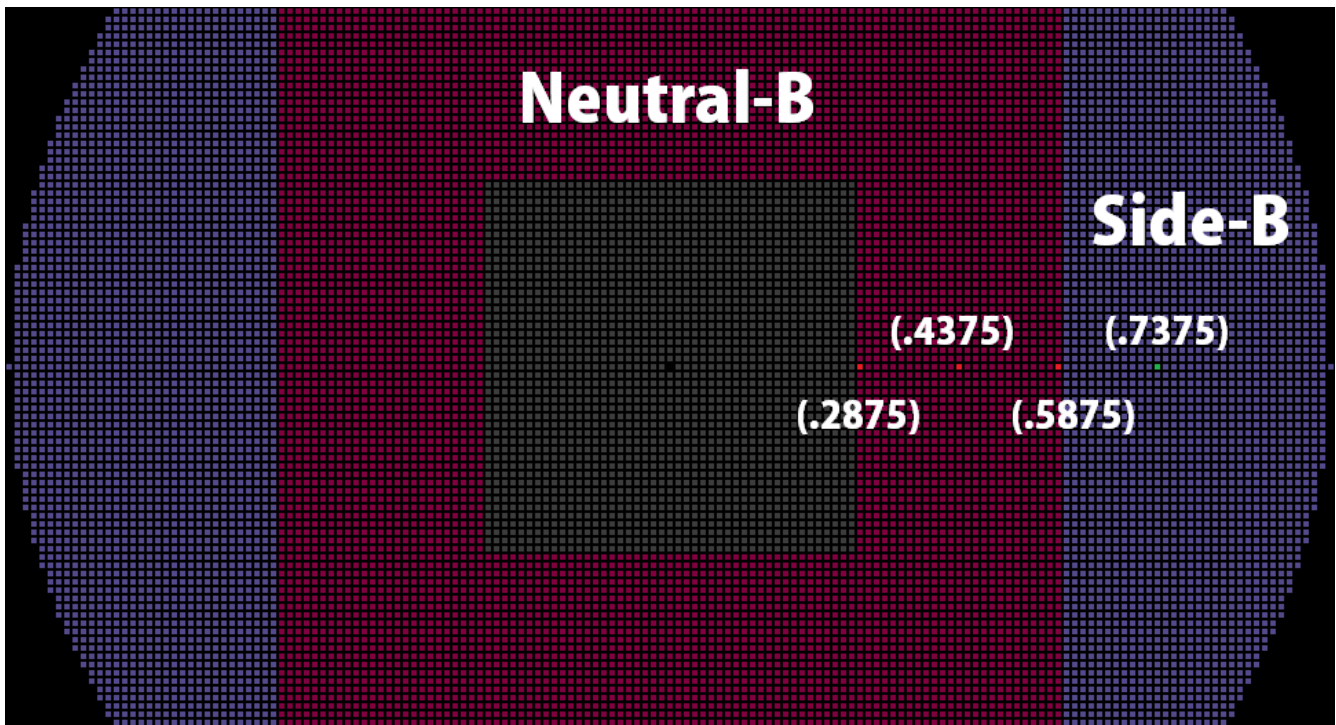
10. Nerfs

Throughout this document, the controller presented to you was deliberately stripped of its ability to perform certain techniques within *Super Smash Bros. Melee* in order to create a fair gameplay experience. These techniques, which require extreme precision to perform with a Gamecube controller, include:

- Ice Climbers desyncs
- Shield Drop directly downwards
- Turning and performing a vertical tilt attack *without* buffering the analog stick in Y-smash
- Perfect ambiguous DI mix-ups when thrown vertically (i.e. X-values directly next to each other instead of spaced out)
- Extremely steep up-B angles
- Extremely steep airdodge angles
- Extremely light analog L/R presses

The B0XX's inability to perform the techniques listed above should have no effect on your in-game decisions so long as you are aware that these techniques are not present on the controller. There are, however, a few additional restrictions on the B0XX that have not yet been documented. This chapter will cover them.

10.1. Accidental Side-B

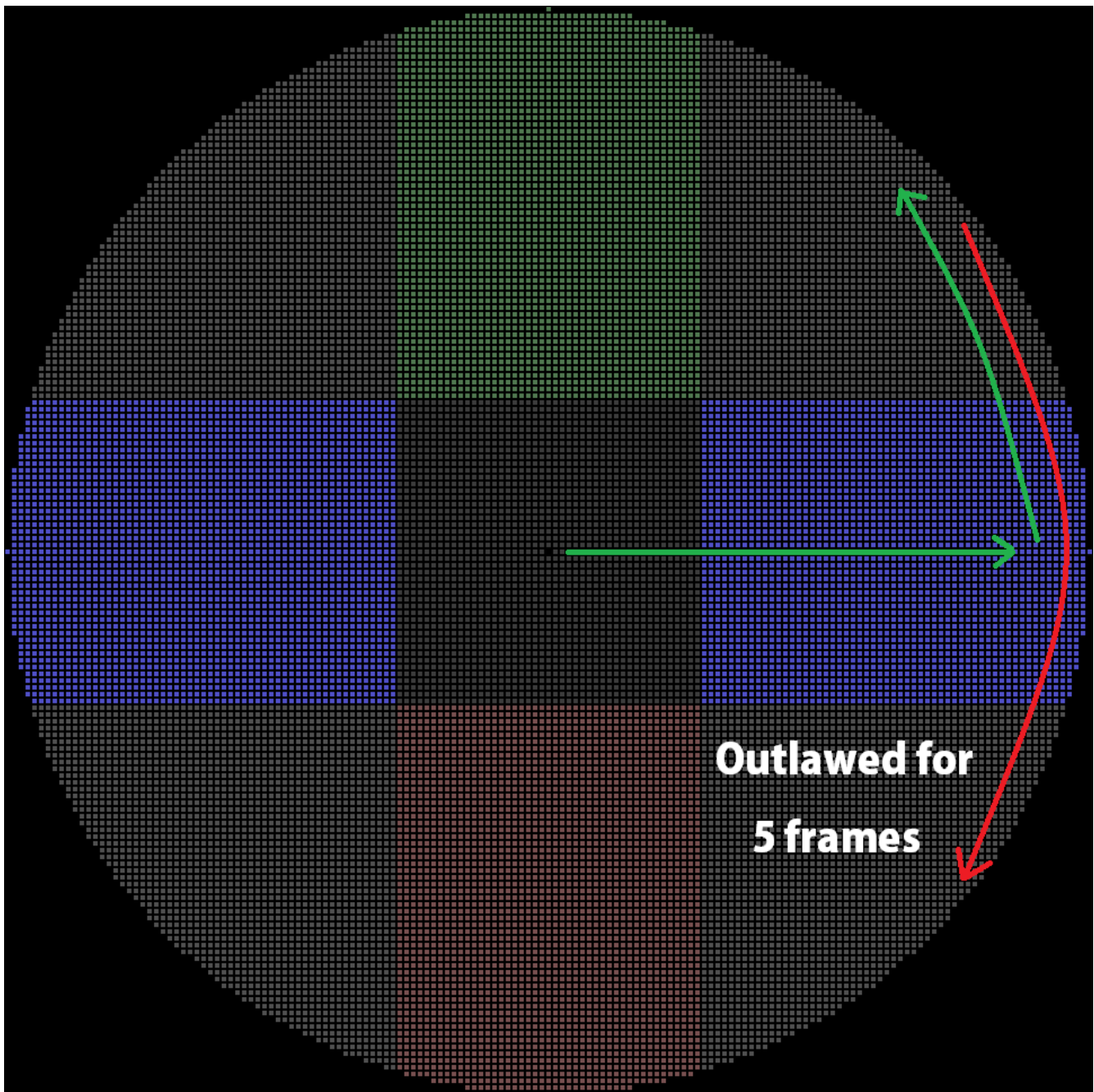


Modifier Y + Horizontal + B: ~~X .2875~~ → X .7375
Modifier Y + Horizontal + A + B: ~~X .4375~~ → X .7375
Modifier X + Horizontal + A + B: ~~X .5875~~ → X .7375

The B0XX contains 3 sets of coordinates that are located in neutral-B territory. While the slight DI options offered by these coordinates are healthy additions to the B0XX, the ability to guarantee a turnaround → neutral-B (without incurring the risk of an accidental side-B) is not.

In order to prevent this exploit, the 3 sets of coordinates pictured in red are modified to X .7375 (a pre-existing coordinate that isn't in neutral-B territory) when the B button is held. **This forces the B0XX user to “flick” the analog stick by quickly pressing, then releasing a horizontal direction in order to perform a turnaround neutral-B.**

10.2. Smash DI



On a Gamecube controller, the go-to method for smash DI is “quarter-circle smash DI,” which involves pointing the analog stick in a cardinal direction followed by an adjacent diagonal direction in order to register 2 smash DI inputs. Due to digital inputs’ lack of physical recoil, however, the B0XX is normally at the advantage of being able to attempt a 3rd smash DI input.

In order to level the playing field, **the B0XX is programmed to recognize a quarter-circle motion on the analog stick and outlaw the correct diagonal direction to go to next for 5 frames in response.** For example:

Frame 1: Right (SDI)

Frame 2: Up + Right (SDI)

Frame 3: Down + Right is outlawed

Frame 4: Down + Right is outlawed

Frame 5: Down + Right is outlawed

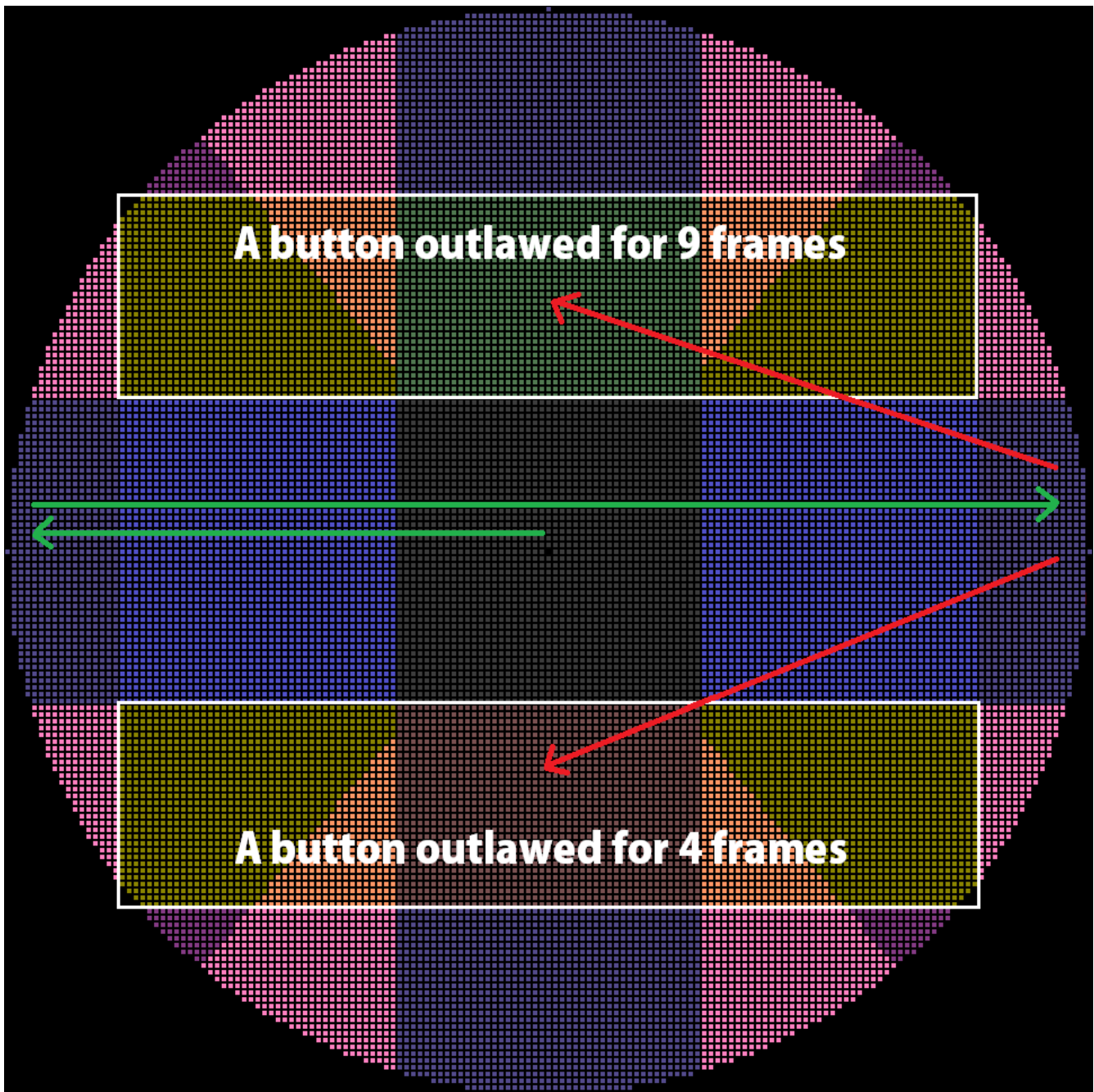
Frame 6: Down + Right is outlawed

Frame 7: Down + Right is outlawed

For clarification, an attempt to input Down + Right during frames 3-7 results in a null input (the analog stick would remain in Up + Right in this case).

Note: The B0XX's smash DI nerf will never conflict with any other area of the game, as the double quarter-circle motion it outlaws is not required to perform any traditional techniques.

10.3. Pivot Tilts



On a Gamecube controller, performing vertical tilts shortly after pivoting is nearly impossible due to the nature of the analog stick motion required. In accordance with this shortcoming, the B0XX is programmed to deny pivot U/UF-tilt attempts for 9 frames and pivot D/DF-tilt attempts for 4 frames.

Note: Only the A button press is outlawed (the directional input is not).